UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

TRABAJO FIN DE MÁSTER

Automatización de funciones en el seguimiento del profesor para la emisión de clases presenciales

Máster Universitario en Ingeniería de Telecomunicación

Alberto Palero Almazán

JULIO 2016

Automatización de funciones en el seguimiento del profesor para la emisión de clases presenciales

AUTOR: Alberto Palero Almazán

TUTOR: Jesús Bescós Cano

Grupo VPULab

Dpto. de Tecnología Electrónica y de las Comunicaciones

Escuela Politécnica Superior

Universidad Autónoma de Madrid

Julio de 2016

Trabajo parcialmente financiado por el Ministerio de Economía y Competitividad del Gobierno de España bajo el proyecto TEC2014-53176-R (HAVideo) (2015-2017)

Agradecimientos

Quiero agradecer a mi tutor, Jesús Bescós, por darme la oportunidad de realizar este trabajo. y por toda su ayuda y asesoramiento. También me gustaría agradecer a Chema toda su ayuda y esfuerzo, involucrándose para que este primer año de máster fuese lo mejor posible.

Gracias a toda la gente del VPU por toda la ayuda y apoyo que siempre me han dado, y por el buen ambiente que hay. En especial me gustaría agradecer a tres personas: a Sara por confiar en que llegaría, aunque yo lo viese imposible, y a Pencho y Rafa por su buen humor y siempre estar dispuestos a echarme una mano, aunque este último tuviese que escapar a EE.UU. para que le dejase tranquilo.

Gracias a Carol por toda su ayuda, consejos e ideas, por dar tanto sin nunca pedir nada a cambio. Gracias también a Diego por todo su apoyo, amenizando las mañanas hablando por Lync.

Gracias a todos mis compañeros de prácticas, no estaría aquí sin vosotros. A Manu, gran inventor del saludo que nunca nos ha fallado. A Sandra, por siempre haber estado ahí y seguir estando después de tantos años. A Álvaro, por ser una de esas personas en las que siempre he podido contar para lo que fuese. A Xan, por ser como es, por apoyarme siempre, incluso desde Irlanda. Ha sido un honor coincidir con todos vosotros.

También quiero agradecer a todos los compañeros de clase, ha sido un verdadero placer pasar todos estos años a vuestro lado.

Finalmente, quiero agradecer a la gente más importante para mí, a mi familia, a mis padres y a mi hermana. No hay palabras que describan su esfuerzo y sacrificio, siempre a mi lado, ayudándome siempre que lo he necesitado. Me lo han dado todo en esta vida, enseñándome con su amor y con su ejemplo. Os estoy eternamente agradecido.

Alberto Palero Almazán

Junio 2016

PALABRAS CLAVE

Algoritmo de seguimiento, filtro de Kalman, emisión de clase, detector de personas HOG, PKLT filter.

RESUMEN

La motivación principal detrás de este trabajo ha sido automatizar el seguimiento del profesor en un aula para emitir clases presenciales a través de internet a estudiantes que, por diversos motivos, no pueden asistir físicamente al aula en que se imparte dicha clase.

Este trabajo es continuación de varios trabajos previos, que como resultado de los cuales se dispone de un algoritmo que a partir de la secuencia de imágenes captada por una cámara fija realiza un seguimiento en tiempo real de la posición del profesor en el aula, y orienta en esa dirección una cámara móvil cuya señal de vídeo es la que finalmente se desea transmitir. Adicionalmente, se dispone de una aplicación web para que los usuarios puedan visualizar las clases y de una aplicación de gestión de este servicio de emisión de clases.

En este trabajo se detalla la integración de un detector de personas HOG, en el algoritmo de seguimiento original con el objetivo de automatizar la etapa de inicialización y, además, mejorar la recuperación del objetivo una vez que el algoritmo determina que lo ha perdido.

Una vez explicadas las modificaciones realizadas en el algoritmo de seguimiento, se detallarán dos nuevas propuestas para mejorar el esquema de reglas original que mueve la cámara móvil y que controla su nivel de zoom, de modo que el efecto sea lo más parecido posible a un cámara que graba la escena. Una de las soluciones está basada en el esquema de reglas original, mientras que la otra intenta predecir la posición futura del profesor mediante el filtro de Kalman.

Por último, se ha evaluado, tanto la inicialización del algoritmo, como la recuperación del objetivo y el control del movimiento de la cámara. Con ello se han sacado las conclusiones apropiadas.

KEYWORDS

Tracking algorithm, Kalman filter, class broadcasting, HOG people detector, PKLT filter.

ABSTRACT

The main motivation behind this Project has been to automate the process of tracking a teacher in a classroom so as to be able to broadcast classes via the internet to students who, for various reasons, can't physically attend the class that is being taught.

This project is a continuation of several previous works, as a result of which there is an algorithm that uses images from a video sequence taken by a fixed camera to track, in real time, the position of a teacher in a classroom, and guides in that direction a mobile camera whose video signal is the image the student will see. Furthermore, there is a web application that allows users to view these classes.

The integration of an HOG people detector in the original algorithm is detailed in this project. This detector is useful so as to be able to automate the initialization phase, and it allows a faster recovery of the target once the algorithm determines that it has lost it.

Once the changes that have been made to the tracking algorithm have been explained, two new proposals that improve the mobile camera movements are introduced. With these proposed solutions better movement control and zoom levels are obtained, making seem as if a person is controlling the mobile camera. Of these approximations, one is based on the original rule scheme, while the other attempts to predict the target's future position using a Kalman filter.

Finally, the initialization of the algorithm, the recovery of the target and the mobile camera´s movement control have been evaluated, and the appropriate conclusions have been drawn.

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

Capítulo 1. INTRODUCCIÓN

1.1 MOTIVACIÓN

El acceso a la enseñanza es un bien fundamental deseado por todos, pero el cual tiene el gran obstáculo que es la distancia. La mayoría de clases se imparten de manera presencial, pero a un estudiante, por diversos motivos, le puede resultar imposible desplazarse hasta el aula.

A lo largo de los años este ha sido un problema que se ha intentado solucionar de diversas formas, sin que ninguna solución propuesta haya sido completamente satisfactoria. Entre las primeras soluciones propuestas estaba la grabación de las clases instalando una cámara fija con un campo de visión que abarcara todo el fondo donde se sitúa el profesor, pero esta solución tenía la desventaja que, si el aula era muy amplia y con múltiples pizarras, lo que el profesor escribiese en ellas era ilegible para un estudiante viendo la grabación. Otras de las propuestas fueron contratar un técnico que grabase de manera manual la clase, haciendo el zoom oportuno, o instalando varias cámaras fijas, una que mostrase una visión general del fondo de la clase y las otras enfocasen exclusivamente a las pizarras. Estas soluciones lograban transmitir la clase de manera aceptable, pero teniendo la clara desventaja de que el coste sería mucho mayor.

Este trabajo pretende aprovechar el conocimiento e investigación que se ha hecho a lo largo de los años con algoritmos de seguimiento para automatizar la grabación de una clase sin necesidad de un técnico o una cantidad no razonable de cámaras. Para ello, se continuará a partir de un Trabajo de Fin de Máster de esta misma Escuela [1], el cual, a partir de una secuencia de imágenes captadas por una cámara fija, era capaz de obtener en tiempo real la posición del profesor en el aula y orientar una cámara móvil, cuya señal es la que se desea transmitir, a dicha posición.

1.2 OBJETIVOS

El objetivo principal de este trabajo es automatizar el seguimiento del profesor en un aula para la emisión de clases presenciales. Para ello, este trabajo se basa en el proyecto [1] donde se hace un estudio exhaustivo sobre el algoritmo de seguimiento óptimo, pero no se profundiza en los siguientes temas que deben ser tratados para que el sistema se pueda poner en funcionamiento:

- Automatización de las etapas de inicialización del algoritmo: En el citado TFM la inicialización del algoritmo se hace a mano, lo cual queda descartado si se desea que el sistema sea completamente automático.

- Mejora del esquema de reglas que controlan el movimiento de la cámara móvil: El movimiento de la cámara en el citado TFM es brusco y molesto para el usuario final. Además, la latencia de la red hace que el movimiento de la cámara vaya con retraso respecto a lo que está sucediendo en la escena.

Para lograr el objetivo principal de este trabajo, se dividirá en tareas que lo hagan más manejable:

- **Estudio detallado del estado del arte:** Conocer en qué consisten los algoritmos de seguimiento, analizando las particularidades, ventajas y desventajas de los diferentes tipos que existen.

- **Aprendizaje del algoritmo de seguimiento a utilizar:** Estudio en profundidad del algoritmo de seguimiento que se utilizará, buscando implementar posibles mejoras para el funcionamiento óptimo del algoritmo.

- **Integración de un algoritmo de seguimiento con un detector de personas:** Se analizarán distintas fases en el algoritmo de seguimiento donde un detector de personas ayude a obtener mejores resultados.

- **Implementación de un sistema de producción automática:** Creación de un módulo que estará encargado de determinar la posición de la cámara PTZ en cada momento de manera automática para poder emitir la clase de forma virtual.

- **Evaluación del algoritmo de seguimiento:** Se evaluarán los resultados del algoritmo de seguimiento analizando si las mejoras implementadas aumentan la calidad del resultado final.

- **Evaluación del control de la cámara PTZ:** Una vez evaluada la calidad del algoritmo de seguimiento, se evaluará el módulo que controla el movimiento de la cámara PTZ determinando si la calidad de la producción final es lo suficientemente buena como para poner este sistema en funcionamiento.

1.3 ESTRUCTURA DE LA MEMORIA

La organización de la memoria de este trabajo se compone de los siguientes capítulos:

- **Capítulo 1:** Motivación y objetivos del proyecto y estructura de la memoria.

- **Capítulo 2:** Estado del arte de algoritmos de seguimiento, en especial *PKLT filter*. Estudio del arte del detector de objetos HOG y su posible aportación al algoritmo de seguimiento utilizado. Identificación y breve descripción de la plataforma web.

- **Capítulo 3:** Detección de posibles puntos de mejora en el algoritmo de seguimiento y soluciones propuestas, entre las cuales se incluye una descripción de la integración del detector de objetos HOG con un algoritmo de este tipo.

- **Capítulo 4:** Descripción de las diversas soluciones propuestas para el módulo que controla el movimiento de la cámara PTZ.

- **Capítulo 5:** Evaluación del algoritmo de seguimiento y del control de movimiento de la cámara PTZ.

- **Capítulo 6:** Conclusiones obtenidas tras el análisis de resultados del trabajo y posibles vías para el trabajo futuro.

Capítulo 2. ESTADO DEL ARTE Y CONCEPTOS BÁSICOS

Una vez que se han enunciado la motivación y los objetivos principales del trabajo, se continúa con una explicación breve pero completa del algoritmo existente, seguido de un estudio del estado del arte de algoritmos de seguimiento, entro otros, que serán utilizados en este trabajo.

Una vez hecho eso se hará una breve descripción de los conceptos básicos en los cuales se basa este trabajo.

2.1 CONTEXTO DE LOS DESARROLLOS

El Trabajo de Fin de Máster [1], en el cual está basado este proyecto, se tenía como objetivo crear un sistema de seguimiento que, originalmente, estaba planeado para ser realizado con una cámara móvil. Es por ello que se implementaron, en un principio, algoritmos de seguimiento que funcionaban tanto con cámaras fijas, como con cámaras móviles. Más adelante, debido a los problemas de retardo en el posicionamiento de la cámara móvil y a su *frame rate* inestable, se decidió aprovechar la infraestructura existente, haciendo uso de una cámara fija adicional.

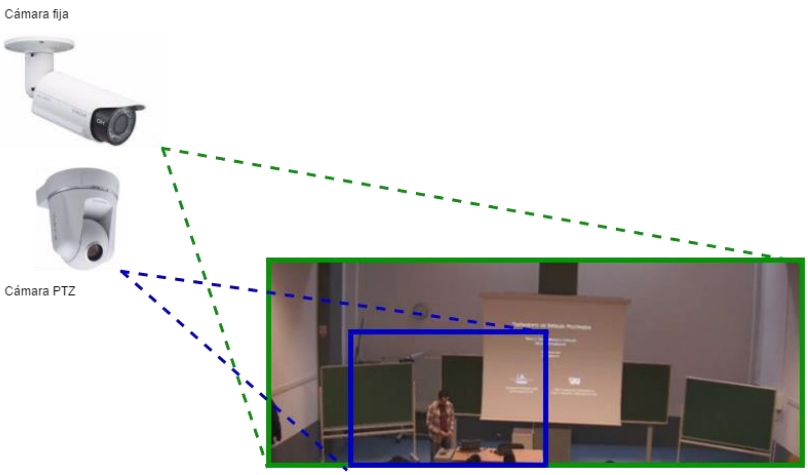


Figura 2.1: Infraestructura de cámaras utilizada.

La imagen final que recibirá el usuario es aquella que se capta con la cámara móvil PTZ, que como se puede ver en la Figura 2.1, tiene un campo de visión menor que el de la cámara fija y, por lo tanto, es necesario realizar una calibración inicial para conocer la posición relativa de ambas y situar correctamente la imagen de una sobre la de la otra. Esta corrección se realiza mediante el cálculo de una homografía, y una vez hecho esto, se pone en marcha el algoritmo, que sigue el diagrama de la Figura 2.2.

En la Figura 2.2 se detalla la ejecución del algoritmo original. Primero de todo, el propio profesor o el técnico que inicialice la aplicación debe seleccionar el objetivo (en

general, al propio profesor que imparte la clase) que se desea seguir, a partir de un *frame* captado por la cámara fija, como se puede observar en la Figura 2.3. A continuación, se genera un modelo basado en histograma que describe al objetivo que se desea seguir. Seguido de eso, se pone en marcha el seguimiento utilizando el algoritmo que mejores resultados dio en [1]. Además, el algoritmo es capaz de detectar cuándo ha perdido al objetivo original y, en caso de haberlo perdido, puede volver a encontrarlo. Finalmente, si el algoritmo determina que no ha perdido al objetivo, se actualiza el modelo de histograma, la posición del objetivo y se manda una instrucción a la cámara PTZ para que se mueva a la nueva posición.

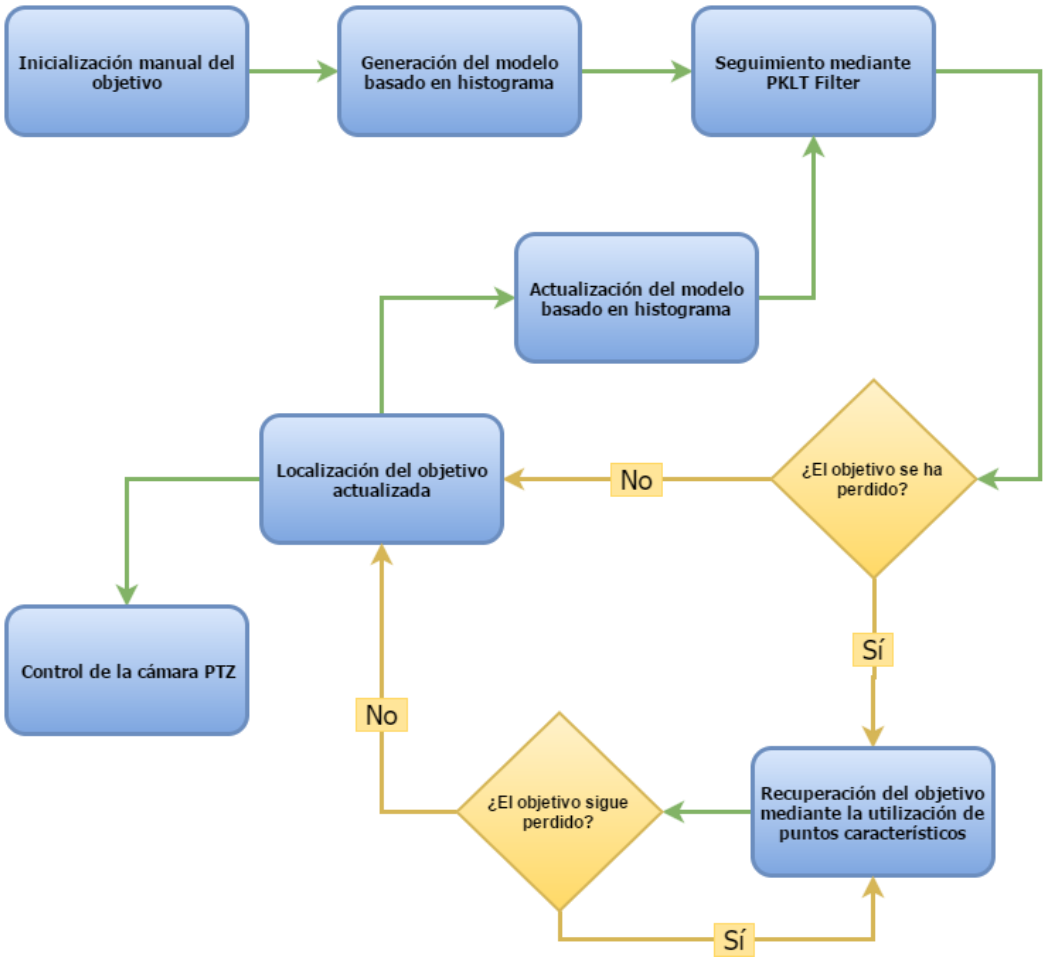


Figura 2.2: Diagrama del funcionamiento del algoritmo de seguimiento original.

Como se ha explicado en el capítulo anterior, en este trabajo se pretende profundizar en la automatización del proceso de seguimiento y por ello, el primer bloque de la Figura 2.2 que se va a modificar es la inicialización del objetivo, utilizando para ello un detector de personas. Además, aprovechando este detector, se intentará mejorar el módulo de recuperación del objetivo. Por último, el bloque de control de la cámara debe ser mejorado ya que el existente, fruto de una aproximación inicial al problema, mueve la cámara de forma muy brusca y puede llegar a ser molesta para el usuario.

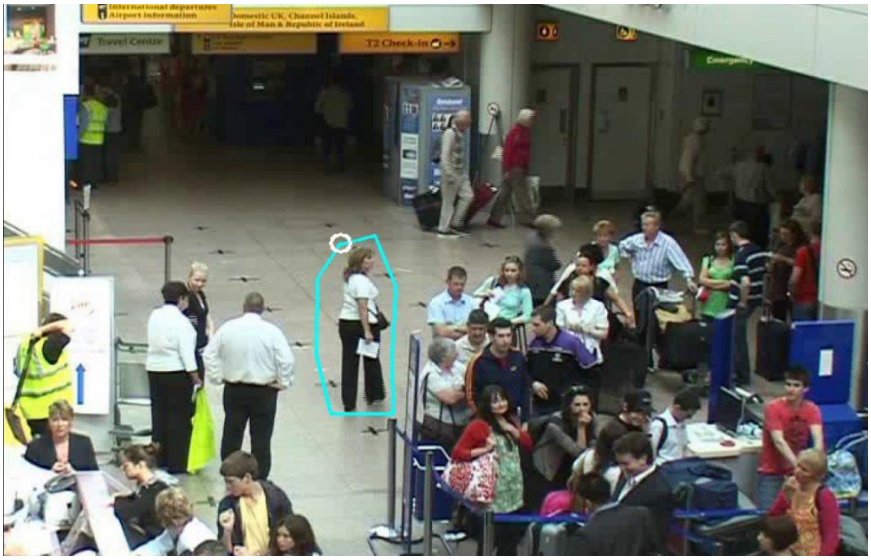


Figura 2.3: Selección manual del objetivo que se desea seguir. Fuente: [1]

2.2 ALGORITMOS DE SEGUIMIENTO

Los algoritmos de seguimiento son precisamente lo que dice su nombre, algoritmos capaces de seguir un objetivo dado, de manera automática, a lo largo de una secuencia de vídeo en una escena determinada. En los últimos años, estos algoritmos han sido de gran interés para la comunidad científica ya que hoy en día se aplican en una gran variedad de ámbitos: vídeo-vigilancia, robótica, conducción automática de vehículos no tripulados, etc.

Si bien es cierto que existen en la literatura multitud de técnicas para hacer seguimiento de objetos, a parte de las explicadas a continuación, el objetivo de este trabajo es hacer un seguimiento de larga duración lo que obliga a descartar algunas soluciones.

Dicho interés en estos algoritmos ha resultado en un gran abanico de soluciones propuestas para cumplir con el objetivo de realizar el seguimiento de forma automática. Una aproximación es la de seguimiento por características mediante KLT (acrónimo de los nombres de los creadores Kanade-Lucas-Tomasi) basado en el trabajo de Lucas-Kanade [2] y posteriormente aclarado pos Shi-Tomasi en [3]. Este método consiste en obtener puntos característicos mediante el método de Shi-Tomasi, que está basado en el detector de esquinas de Harris, y hacer el seguimiento aplicando las ecuaciones desarrolladas por Lucas-Kanade, que a su vez implementan una búsqueda de ascenso por gradiente.

Otra de las técnicas más usadas en los últimos años es la de algoritmos de seguimiento basados en filtros de partículas. Este tipo de filtros derivan de los modelos de Bayes en los cuales la observación del instante actual depende del estado actual, y el estado actual solo depende del estado anterior. De forma resumida, el seguimiento

utilizando este método se basa en repartir un conjunto de puntos de manera aleatoria por la imagen y asignarles valor, a continuación, se vuelve a repartir un nuevo conjunto de puntos por la imagen, que reemplazará a los anteriores, pero asignándoles un valor que dependerá de los anteriores, para posteriormente modificar el estado de cada uno de ellos con la finalidad de utilizar estos para predecir el estado en el instante siguiente. Esta solución basada en los modelos de Monte Carlo fue usada por primera vez en [4] y para una descripción más detallada de estos modelos de seguimiento se puede ver en [5].

Finalmente, otra aproximación, que es popularmente usada en los algoritmos de seguimiento, es la de Mean-Shift, que hace una búsqueda del objetivo mediante esta técnica de ascenso de gradiente. Resumiendo [6], en esta técnica se utiliza la citada técnica para localizar en el *frame* siguiente la ventana más parecida.

2.2.1 PKLT Filter

Como se ha dicho anteriormente, este trabajo está basado en un proyecto de esta misma escuela [1], en el cual se hizo un estudio minucioso de una gran cantidad de algoritmos de seguimiento de larga duración, donde con el que mejores resultados se obtuvieron fue el algoritmo de seguimiento mediante partículas KLT "*PKLT Filter*" y es por ello que se ha decidido continuar con él en este trabajo.

Para poder hacer un seguimiento de larga duración, es necesario un algoritmo que tenga en cuenta distintos tipos de características y que a la vez puedan ser renovadas continuamente, o filtradas en caso de degeneración. Por esa razón, este filtro implementa de manera conjunta la detección y seguimiento de características mediante el uso de las KLT y se aplican los principios de actualización y filtrado de los filtros de partículas, usando cualquier tipo de información (movimiento, color, forma, etc.), aunque en este trabajo solo se utiliza la información de color y de movimiento.

En la Figura 2.4, se puede observar el diseño de este sistema de seguimiento, donde se puede observar que se realiza una búsqueda de partículas y un filtrado por movimiento y por degradación.

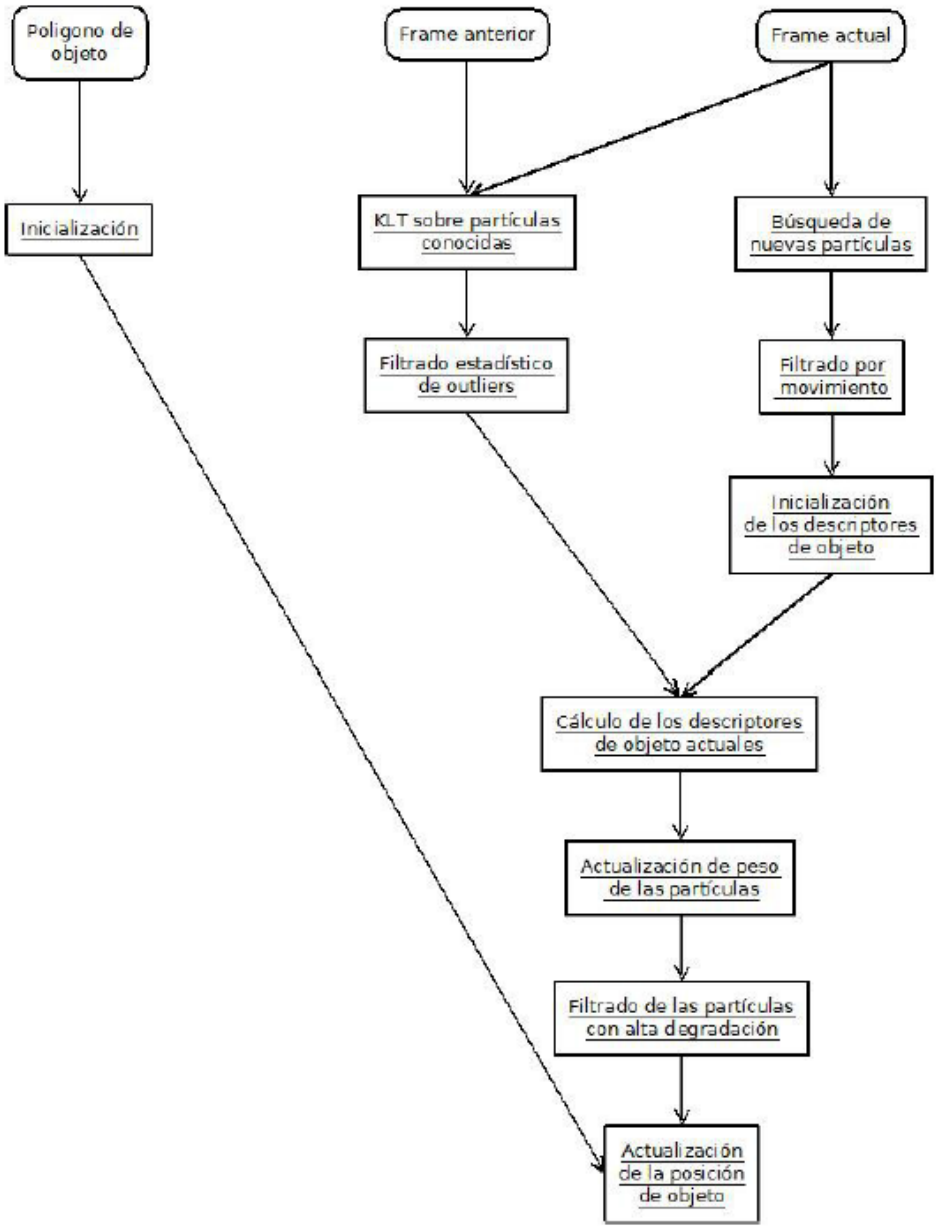


Figura 2.4: Sistema de seguimiento PKLT. Fuente: [1]

2.3 DETECCIÓN DE PERSONAS CON HOG

Tal y como se ha mencionado en la sección 1.2, uno de los objetivos principales de este trabajo es inicializar de manera automática el algoritmo de seguimiento, de forma que éste siga a lo largo de una secuencia de vídeo al profesor, ya que en [1] la inicialización se hacía a mano, dibujando un polígono alrededor de lo que se desease seguir.

En general, los detectores de personas consisten en el diseño y el entrenamiento de un modelo de persona basado en parámetros característicos de la misma. En particular, el detector de personas HOG (acrónimo de la expresión inglesa *Histogram of*

Oriented Gradients) se basa en la evaluación de histogramas locales y normalizados de las orientaciones de los gradientes de una imagen. La idea principal detrás de este detector es que se puede caracterizar la forma y la apariencia de un objeto mediante la distribución de intensidad local de los gradientes como por la dirección de los bordes.

Para realizar dicha detección, la imagen es dividida en pequeñas regiones conectadas llamadas celdas, y para cada pixel dentro de cada celda se genera un histograma de la dirección del gradiente. Finalmente, dichos histogramas se concatenan para crear el detector.

a) b) c)

Figura 2.5: a) Imagen de entrada. b) Promedio del gradiente. c) Descriptor HOG. Fuente: [8]

Para una precisión mayor, los histogramas locales se pueden normalizar calculando la intensidad en una región mayor de la imagen, llamada bloque, y utilizar este valor para normalizar todas las celdas dentro del bloque. Dicha normalización resulta en una mayor invariancia del detector a sombras y cambios de iluminación en la escena.

2.4 FILTRO DE KALMAN

Para la parte del control del movimiento de la cámara hay que tener en cuenta que existe cierto retardo en la red, por lo que desde el instante en el que se manda la instrucción a la cámara, para que se mueva a la nueva posición, hasta que la cámara apunta a dicha posición, transcurre un tiempo en el cual le da tiempo al profesor a volver a moverse, lo que causa que sea posible que el profesor desaparezca del campo de visión de la cámara PTZ. Es por esta razón que se ha probado a anticipar la posición del objetivo utilizando el filtro de Kalman [9].

El filtro de Kalman es un algoritmo que utiliza una serie de medidas observadas a lo largo del tiempo, que se asume que tienen ruido estadístico y otro tipo de inexactitudes, y produce una única estimación, de variables desconocidas, que tiende a ser más precisa que una basada en una única medida.

La estimación del filtro de Kalman se hace utilizando un control de realimentación, es decir, estima el proceso en un instante *t,* y entonces recibe retroalimentación por medio de los datos observados. Esto quiere decir que las ecuaciones que implementa el filtro de Kalman se pueden dividir en dos grupos; aquellas que realizan la predicción y aquellas que se encargan de actualizar la predicción.

Las ecuaciones que realizan la predicción se basan en el estado en el instante anterior, *t-1*, para predecir el estado en el instante actual, *t*. Las ecuaciones del segundo grupo son las responsables de la retroalimentación, es decir, son las encargadas de mejorar la estimación obtenida, mediante la incorporación de nueva información a la estimación anterior. Por estos motivos, el filtro de Kalman se puede ver como un algoritmo de pronóstico-corrección.

2.5 PLATAFORMA WEB

El objetivo principal de este trabajo automatizar el seguimiento del profesor en un aula para la emisión de clases presenciales. Es por ello que, una vez automatizado el proceso de la inicialización del algoritmo de seguimiento y el control de la cámara PTZ, es necesario implementar alguna plataforma para poder emitir la clase a aquellos estudiantes que, por diversos motivos, les puede resultar imposible desplazarse hasta el aula.

Figura 2.6: Ejemplo de lo que vería un estudiante.

Para realizar la emisión de las clases se utilizará una plataforma web diseñada en un Proyecto de Fin de Carrera de esta misma Escuela [11]. En dicho trabajo se crea un sistema pensado para emitir clases presenciales en directo a través de Internet. El funcionamiento de esta plataforma web es muy sencilla e intuitiva de utilizar, primero un estudiante se registra en ella y a continuación el administrador le da de alta en aquellas asignaturas en las que se encuentre el estudiante matriculado. Una vez completado el registro, el estudiante solo ha de entrar en la página web a la hora en la que se imparta la clase y podrá visualizarla (ver Figura 2.6) sin necesidad de estar físicamente en el aula.

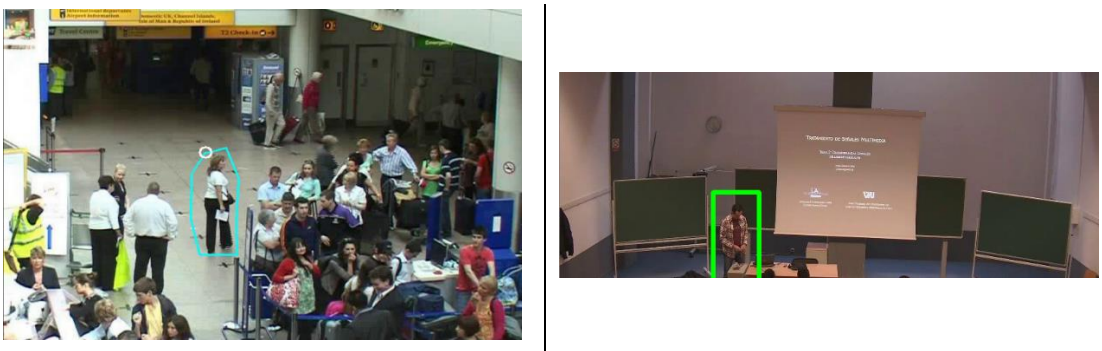
Capítulo 3. MEJORAS INTRODUCIDAS EN EL MÓDULO DE SEGUIMIENTO

Como se ha dicho en el capítulo 1 de esta memoria, el objetivo principal de este trabajo es automatizar el seguimiento del profesor en un aula para la emisión de clases presenciales, basándose en el proyecto [1], y usando el algoritmo de seguimiento que mejor resultados dio.

Al utilizar dicho algoritmo de seguimiento se vio que existían ciertos puntos donde era posible mejorar su funcionamiento, y es en este capítulo se describen y detallan las mejoras incluidas en el módulo de seguimiento, de manera secuencial a la ejecución.

3.1 INICIALIZACIÓN AUTOMÁTICA CON HOG

Como se ha explicado en la sección 1.2, el algoritmo de seguimiento de [1] se inicializaba a mano, donde el propio profesor o un técnico era el encargado de definir la zona que se deseaba seguir, como se puede ver en la Figura 3.1a), donde el recuadro azul es la zona a seguir seleccionada.



a) b)

Figura 3.1: Selección del objeto sobre el que se desea realizar el seguimiento: a) de forma manual. Fuente: [1]. b) de forma automática.

En este trabajo, se requiere que el sistema sea completamente automático y que no sea necesario tener a un operario seleccionando el objetivo siempre que se quiera emitir una clase, obteniendo el resultado de la Figura 3.1b) de manera automática. Es por este motivo que una primera aproximación fue la integración en el algoritmo, del detector de objetos HOG, para detectar al profesor al comienzo de la emisión. Esto era posible dado que se asume que el profesor es la única persona presente en la escena.

Para hacer esto, se sustituyó la parte del algoritmo donde se seleccionaba al objeto a seguir, por un módulo que recibe el *frame* actual y en él se busca al profesor, devolviendo una *bounding box* conteniéndole, en caso de encontrarlo. Esta solución tuvo el gran problema que, debido al funcionamiento del HOG, si el umbral de detección era demasiado laxo, el detector no devolvía solamente la posición del profesor, sino que podía identificar otras zonas de la escena como personas. Esto último ocurría sobre todo

provocado por la forma de las pizarras, donde el detector llegaba a confundir las patas con piernas y el borde inferior con una cintura, dando lugar a una detección errónea (véase Figura 3.2).

a)

b)

Figura 3.2: Resultados obtenidos implementando el detector de personas HOG con: a) umbral laxo. b) umbral restrictivo.

Por el contrario, el umbral se puede hacer más restrictivo, buscando así que el detector solo obtenga la posición del profesor. El problema que tiene esta solución es que, aun así, no se garantiza que se detecte una persona, ni que cuando solo haga una detección, ésta sea el profesor. Además de esto último, poner un umbral tan restrictivo provoca que en la mayoría de las detecciones no se obtenga ninguna detección, como se puede ver en la Figura 3.2. Finalmente, también se intentó utilizar los pesos asociados a cada detección que devuelve el detector HOG implementado en OpenCV, pero dichos pesos no siempre eran fiables por lo que se decidió buscar otra solución a este problema.

La siguiente aproximación para automatizar la inicialización del algoritmo de seguimiento se basó en la descrita anteriormente, ya que la utilización del HOG con un umbral laxo detectaba, la gran mayoría de veces, al profesor, aunque también zonas donde no había persona alguna. Por lo tanto, se trató de idear alguna manera en la que distinguir la *bounding box* que delimitaba al profesor de las demás.

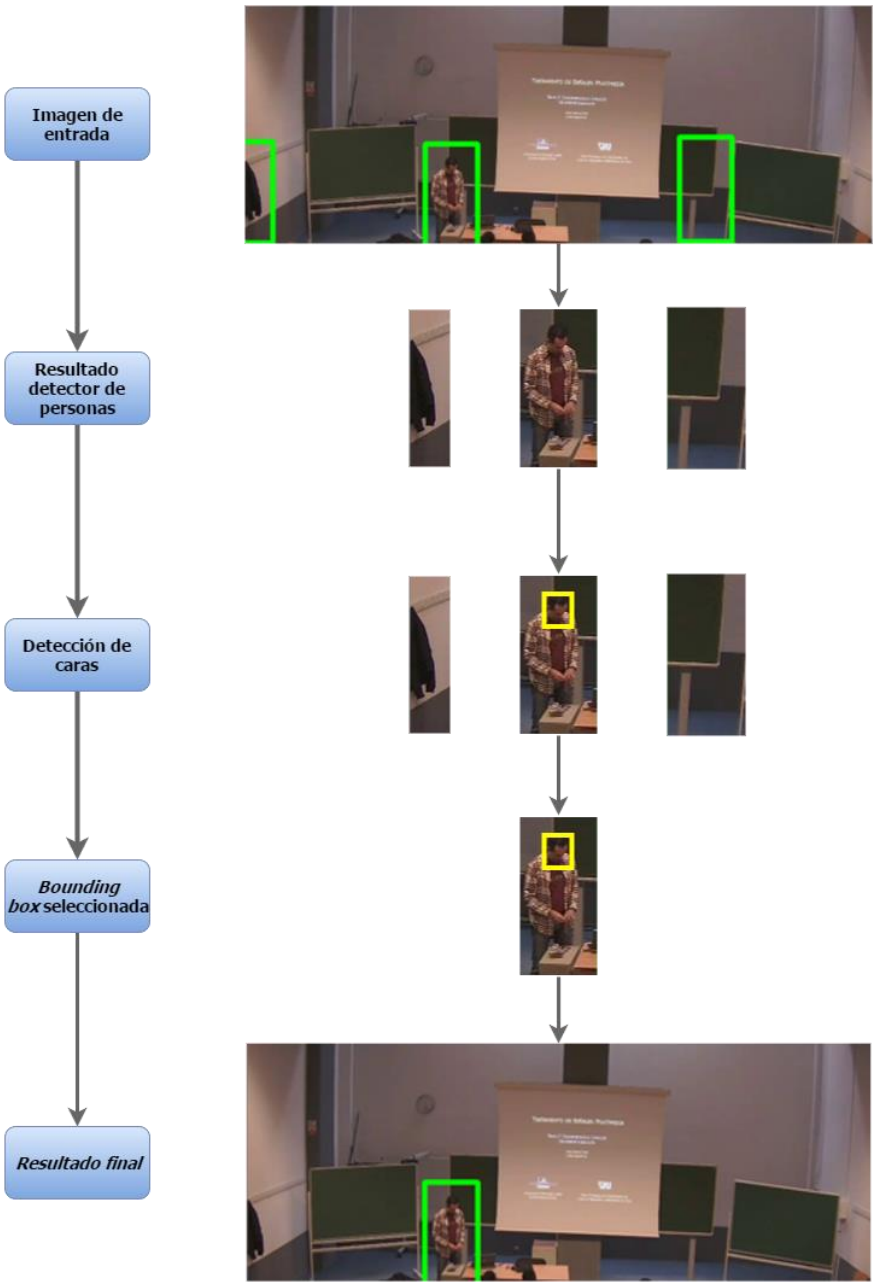


Figura 3.3: Módulo para la inicialización automática del algoritmo.

Para poder distinguir entre el profesor y los falsos positivos que se obtienen con el detector de personas HOG, se ha utilizado el clasificador en cascada, que implementa OpenCV, para detectar caras frontales en las regiones donde el detector de personas dice que hay una persona.

Como se puede ver en la Figura 3.3, se recibe la imagen de la escena y se ejecuta, con un umbral laxo, el detector de personas sobre esa imagen, obteniendo, en este ejemplo, varias detecciones. A continuación, se recortan las regiones donde, según el detector HOG, hay personas, y se ejecuta sobre cada una de ellas el detector de caras, de manera que, si se detecta al menos una cara, dicha región es la que se considera que contiene al profesor.

Durante las pruebas que se hicieron para comprobar el funcionamiento de aplicar el detector de caras a la región que se obtiene con el detector de personas, se vio que había ciertas imágenes en las que no se llegaba a detectar una persona debido a que la persona no miraba de frente, sino que miraba a un lado. Es por ello que además del detector de caras mencionado en el párrafo anterior, se incluyó un detector de caras de perfil, aumentando así las posibilidades de detectar al profesor, ya que basta con que uno de los detectores de caras, ya sea el de cara frontal o el de cara de perfil, obtenga un resultado para considerar que la región analizada contiene al profesor.

Cabe mencionar que los detectores de caras tampoco son perfectos, y existe la posibilidad de obtener falsos positivos y, por lo tanto, dos regiones del detector de personas pueden tener caras. En este caso, para decidir qué región es la que realmente contiene al profesor, se utilizan los pesos que devuelve el HOG, de manera que nos quedamos con aquella que tiene más peso.

3.2 CORRECCIÓN DEL MODELO CBWH

Una vez obtenida la región que contiene al profesor, se procede a crear el histograma, que describe dicha región, utilizando las características de color (rojo, verde y azul) y el indicativo de borde. Las características de color del histograma, llamado histograma RGBE por las características utilizadas, están cuantificadas de [0, 16] cada una y el indicativo de borde es binario, por lo que se obtiene un histograma de 16 x 16 x 16 x 2 = 8192 elementos.

En el algoritmo original de [1], la inicialización se hacía a mano, pudiendo así recortar, única y exclusivamente, al objeto o persona que se desease seguir por la escena. El problema que se dio en este trabajo es que la región obtenida con el detector de personas contiene parte del fondo además del profesor (véase Figura 3.4), y por lo tanto el histograma se creará, no solo con información del objetivo que se desea seguir, sino que también con la información del fondo. Esto provoca que, a la hora de hacer el seguimiento, cuando se hace la comparación del histograma, la disimilitud entre histogramas aumente.

Figura 3.4: Región que interesa seguir, dentro de la región devuelta por el detector de personas.

Para solucionar este problema se buscaron diferentes técnicas para recortar únicamente al objetivo, y crear así un histograma mucho más ajustado. Finalmente, se optó por usar una técnica que ya había sido utilizada en un trabajo de esta misma Escuela [13], que intenta corregir el histograma de una imagen que contiene fondo, que es precisamente lo que buscamos. Dicha técnica llamada CBWH (acrónimo de la expresión inglesa *Corrected background-weighted histogram*), explicado en detalle en [12], intenta corregir el histograma a partir de otro del doble de tamaño que, por lo tanto, contiene más fondo.

A continuación, se intentará explicar de manera visual en qué consiste la técnica CBWH. Primero de todo, una vez que se tiene la *bounding box* que contiene al profesor, se calcula su histograma RGBE, que denominaremos como histograma de FG (siglas de la palabra inglesa *foreground*). Hecho eso, nos generamos una segunda *bounding box* centrada en el mismo punto que la anterior, pero con el doble de altura y ancho, para finalmente, y de la misma manera, calcular su histograma, que dominaremos histograma de BG (siglas de la palabra inglesa *background*).

En la Figura 3.5 se puede observar una ilustración de lo que podría ser el histograma normalizado de FG en verde, donde el máximo global corresponde al objetivo y el segundo pico corresponde al fondo que hay dentro de la *bounding box*. En azul, de esa misma figura, podemos observar el histograma normalizado de BG, en el cual el fondo es más recurrente y, por lo tanto, donde antes teníamos el segundo pico, correspondiente al fondo, ahora es máximo global.

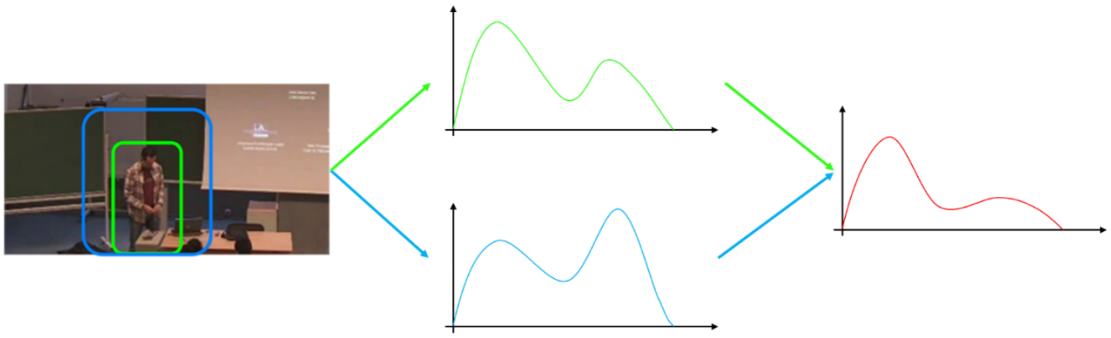


Figura 3.5: Ilustración de los histogramas de FG inicial (verde), BG (azul) y FG final (rojo).

Obtenidos los histogramas, se procede a corregir el histograma de FG. Para ello, se invierte el histograma normalizado de BG y se multiplica por el histograma normalizado de FG, de manera que, como se puede ver en la Figura 3.5, el pico que en el histograma original de FG representaba la zona de fondo, se ve reducido, manteniendo el máximo global en la zona que representa al profesor. Esto es debido a que el peso de fondo en el histograma de BG es mucho mayor que el del profesor, por lo tanto, al multiplicar por la inversa, se ve mucho más reducida la zona de fondo en el histograma de FG final.

3.3 ACTUALIZACIÓN DEL MODELO

Otra pequeña modificación que se introdujo en el algoritmo de seguimiento de [1] fue limitar la actualización del histograma RGBE que describe al objetivo que se desea seguir. La gran mayoría de algoritmos de seguimiento, y sobre todo los algoritmos de seguimiento de larga duración, tienen un módulo que actualiza el modelo del objetivo a la par que se está siguiendo al objetivo. Esta actualización del modelo es necesaria para adaptarse a posibles cambios en la escena y en el propio objetivo que pueden ser causados por todo tipo de razones, la más común siendo un cambio de iluminación.

Por ello, para que el algoritmo de seguimiento de secuencias de larga duración, se adaptase a cambios en la escena a lo largo del tiempo, originalmente, actualizaba el modelo por cada nuevo *frame* que procesaba, si en dicho *frame* se encontraban puntos característicos cercanos a la última posición del objetivo. Esta actualización se decidió limitar para hacerlo solo en aquellos casos en los cuales se considera bastante probable que la región detectada siga siendo el objetivo que se escogió en la inicialización.

Para determinar si la región es parecida, o no, a la del modelo, se obtiene el histograma RGBE de la nueva región y se compara con el histograma del modelo calculando la distancia de Bhattacharyya, obteniendo así un valor entre cero y uno, que representa la disimilitud entre los histogramas. Cuanto más cercana a cero sea la distancia, más similares son los histogramas, por el contrario, cuanto más cercana a uno, más disímiles son.

A pesar de introducir esta restricción a la hora de actualizar, se ha decidido mantener la forma en la que se hace dicha actualización del modelo. Dicha técnica de actualización no consiste en calcular un nuevo histograma, sino que, se crean ventanas de 11x11 alrededor de los puntos característicos que están cerca de la última posición conocida del objetivo, y con dichas ventanas se actualiza el histograma RGBE.

3.4 RECUPERACIÓN DEL OBJETIVO CON HOG

El algoritmo de seguimiento puede perder al objetivo que está siguiendo por diversos motivos, algunos ejemplos pueden ser; que el objetivo salga del campo de visión de la cámara fija, que es a partir de la cual se hace el seguimiento, o el profesor puede irse detrás de alguna pizarra, perdiéndole de esta manera por obstrucción.

Como se ha dicho anteriormente, el algoritmo tiene que funcionar, obligatoriamente, en secuencias de larga duración y, por lo tanto, debe ser capaz de recuperarse de estas situaciones y encontrar al objetivo cuando vuelva al campo de visión de la cámara. Para ello, el algoritmo incorpora un módulo cuya función es encontrar al objetivo una vez que se ha detectado que ha dejado de seguir al objeto.

Para que el algoritmo considere que ha perdido al objetivo comprueba si la disimilitud entre el histograma actual y el del modelo supera cierto umbral, o si en un número consecutivo de *frames* no ha tenido puntos característicos. Los puntos característicos se obtienen sólo de aquellas regiones de la imagen donde hay movimiento, por lo tanto, como se está siguiendo a un profesor en un aula, es posible que el profesor se quede quieto y el algoritmo determine que ha perdido al objetivo.

Originalmente, una vez que el algoritmo determinaba que había perdido al objetivo que estaba siguiendo, realizaba una búsqueda exhaustiva de puntos característicos en la imagen, y se quedaba con aquellos que estuviesen en zonas donde había movimiento, provocando así, que el profesor no pudiese ser encontrado hasta que se moviese y se encontrase en el algún punto característico.

El problema descrito en el párrafo anterior motivó a que se buscase una manera sencilla de solucionar dicho problema. Para ello, visto los buenos resultados que dio en la inicialización, se optó por usar el detector de objetos HOG, junto con los detectores de caras frontales y de perfil, para detectar al profesor. El funcionamiento es el mismo que el descrito en la sección 3.1, el módulo recibe una imagen sobre la cual se lanza el detector de personas, a continuación, sobre los resultados que se obtienen, se lanzan los dos detectores de caras para así quedarse con la *bounding box* que contiene una persona.

Dado que el resultado obtenido puede no ser el objetivo original que se estaba siguiendo, se calcula su disimilitud con el histograma del modelo para corroborar que, efectivamente, la detección es el profesor.

En la inicialización del algoritmo, los tiempos de ejecución del detector de personas y de caras no son importantes dado que, hasta que no se inicializa, no comienza a realizar el seguimiento, y por lo tanto no afecta a nada. En cambio, cuando se quiere recuperar al objetivo, se desea que este proceso sea lo más rápido posible, de manera que se minimice el tiempo en el cual el algoritmo no está siguiendo al objetivo. Por este motivo, y debido a que los detectores no son capaces de trabajar en tiempo real, se ejecuta la recuperación del objetivo mediante detectores cada cinco *frames*, de manera que no se detiene la ejecución del algoritmo de forma notable.

Capítulo 4. MOVIMIENTO DE LA CÁMARA PTZ

Una vez que se han explicado las modificaciones que se han introducido al algoritmo de seguimiento original, se va a detallar el módulo de control de la cámara PTZ, que será la imagen que finalmente visualice el usuario.

En [1], el movimiento de la cámara era muy brusco e incluso llegaba a ser molesto, y es por esa razón que se tuvo que replantear este sistema de producción. En este capítulo se detallan dos posibles soluciones que se han puesto en marcha y probado, con el objetivo de mejorar este módulo.

4.1 MOVIMIENTO UTILIZANDO REGLAS

El módulo original que controlaba el movimiento de la cámara funcionaba a partir de un conjunto muy sencillo de reglas, siguiendo el esquema de la Figura 4.1. Cada cuatro *frames* se obtenía la última posición conocida del objetivo que se estaba siguiendo y, además, la posición actual de la cámara PTZ. Restando las dos posiciones, se obtenía el desplazamiento del objetivo respecto a la posición actual de la cámara.

Una vez conocido el desplazamiento, se hacían dos cosas; primero, se comprobaba si el movimiento, en píxeles, en el eje horizontal o vertical superaban un umbral, que era un cinco por ciento de la imagen en horizontal o en vertical respectivamente. Si dicho umbral se superaba, se mandaba una instrucción a la cámara con la nueva posición, en caso contrario no había comunicación con la cámara. Después de realizar esto, lo que se hacía era actualizar una variable que indicaba el zoom que debía de tener la cámara, incrementándolo en caso de que no hubiese movimiento o, en caso contrario, reduciéndolo.

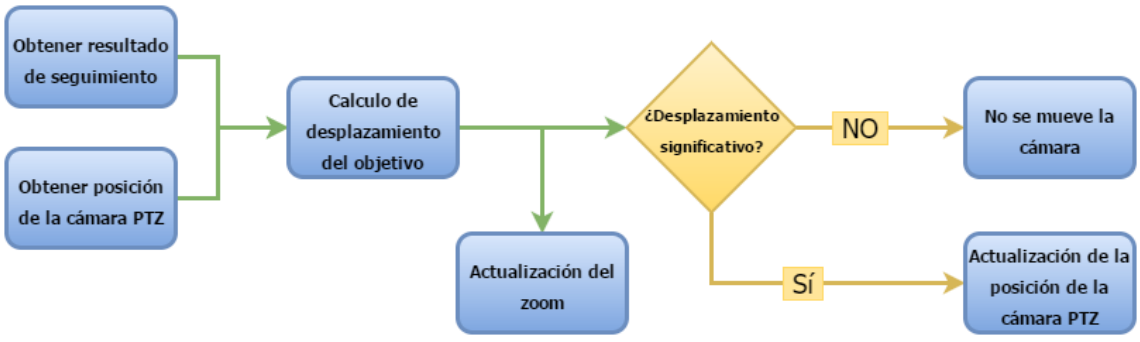


Figura 4.1: Diagrama de reglas original.

El planteamiento descrito anteriormente tenía varios defectos que hicieron que se tuviese que replantear el módulo de control de la cámara PTZ. El primer error era que el zoom de la cámara se actualizaba después de haberla movido, teniendo que esperar a que el objetivo se moviese para que dicha modificación del zoom tuviese efecto. Otro

problema era que no se tenía en cuenta si el algoritmo de seguimiento había perdido al objetivo o no, de forma que, si se había perdido, la posición que devolvía el algoritmo de seguimiento no se actualizaba y por lo tanto la cámara PTZ se quedaba en una posición que no debía estar. Además, esto último también provocaba que, al no haber movimiento, el zoom se fuese incrementando y, por lo tanto, cuando se volviese a encontrar al objetivo y se moviese la cámara, lo haría con un zoom excesivo.

Además de los problemas descritos en el párrafo anterior, existían otros dos problemas con ese modelo, pero más fáciles de solucionar. Cuando la cámara recibía una instrucción, esta se bloqueaba, no admitiendo más instrucciones hasta completar la que estaba realizando: por ello la velocidad a la que se movía la cámara se había situado al máximo, de forma que se adaptase lo más rápidamente posible a movimientos del objetivo. El problema era que, al poner una velocidad tan rápida, se obtenían unos movimientos muy bruscos y molestos para el usuario.

El segundo problema, de fácil solución, era provocado por los umbrales que decidían si el objetivo se había movido lo suficiente como para mover la cámara. Al ser la escena un aula, es plana y, por lo tanto, a no ser que el zoom fuese muy alto, el movimiento vertical no debería tener tanta importancia como un movimiento horizontal. Esto, junto con la velocidad a la que se movía la cámara provocaba que el movimiento no pareciese natural, y fuese un movimiento brusco y repentino.

La primera solución propuesta para el control de la cámara móvil es el que se muestra en la Figura 4.2. Se puede apreciar que es similar a la que ya existía en [1], ya que está solución también hace uso de un conjunto de reglas para controlar el movimiento de la cámara móvil y, además, se ejecuta cada cuatro *frames*.

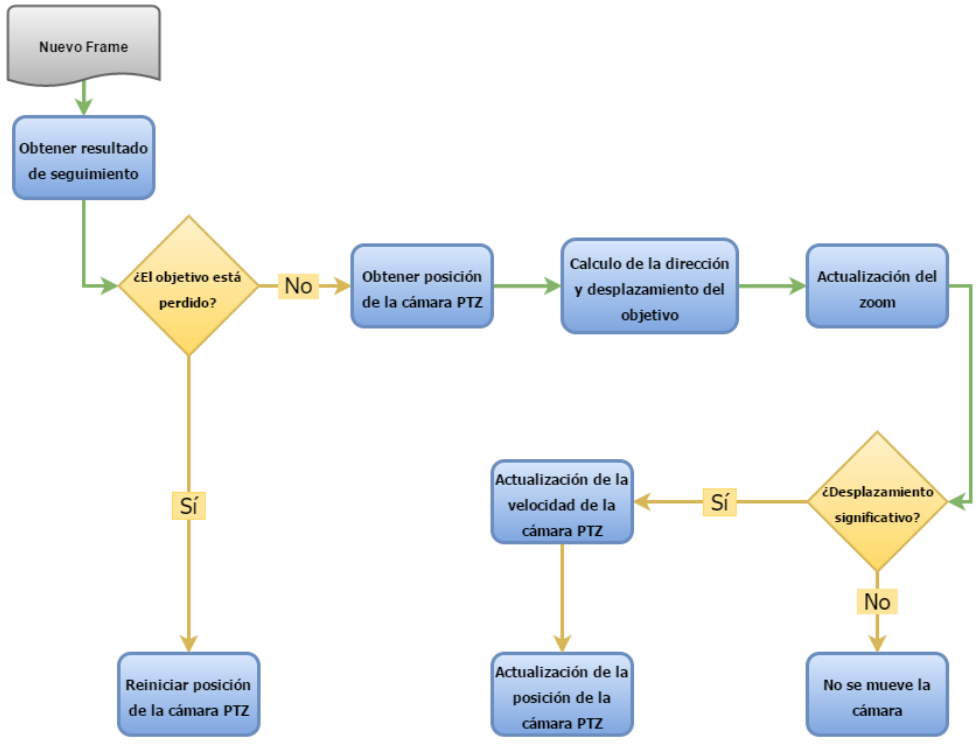


Figura 4.2: Diagrama propuesto para el control de la cámara basado en reglas.

La primera diferencia entre la solución propuesta y la que existía es la comprobación de si el algoritmo ha perdido, o no, al objetivo antes de comenzar el proceso. Aquí también se incluye otro cambio importante, si el algoritmo ha perdido al objetivo, en lugar de mantener la cámara en su posición, se reinicia su posición, de manera que apunta al centro de la escena con el mínimo zoom posible, de forma que, aún sin saber la posición del objetivo, se pueda ver toda la escena.

A continuación, utilizando la posición que devuelve el algoritmo de seguimiento y conociendo la posición actual de la cámara se obtiene el desplazamiento del objetivo respecto a la posición de la cámara y, además, también se calcula en qué dirección se está moviendo. Esto último, obtener la dirección del movimiento, se hace para saber cuándo cambia de dirección, y así poder parar la cámara.

Luego, se calcula el zoom que debería tener la cámara, explicado en la sección 4.3, para, a continuación, comprobar si hay suficiente movimiento como para moverla. Dicha comprobación es distinta a la que existía anteriormente, en este caso se comprueba si el objetivo se ha movido, del tamaño en píxeles de la imagen captada por la cámara fija, al menos un cinco por ciento en horizontal o un quince en vertical. En caso de no cumplirse la comprobación, y se considere que no hay suficiente movimiento, la posición de la cámara no se modifica. Por el contrario, si se considera que sí hay suficiente movimiento, se calcula la velocidad a la que se debería mover la cámara, utilizando la posición de la misma cámara y del objetivo.

El controlar la velocidad a la que se mueve la cámara se hace para obtener un mayor control, consiguiendo así que el movimiento sea suave y fluido. Tras muchas pruebas, se vio que, de las 24 velocidades disponibles a las que se puede mover la cámara, las cinco más lentas eran las más agradables visualmente, y las demás llegaban a ser molestas. De las cinco velocidades, el algoritmo utiliza la que corresponda en cada momento dependiendo de la distancia que hay entre la posición de la cámara y la del objetivo, a mayor distancia, mayor velocidad y viceversa.

4.2 MOVIMIENTO UTILIZANDO EL FILTRO KALMAN

La solución propuesta en la sección anterior tiene un gran punto débil que es el retardo que existe desde el instante en el que se manda la instrucción para mover la cámara hasta que se ejecuta dicho movimiento. Dicho retardo es tan grande que es posible que el objetivo, si se mueve demasiado deprisa, salga del campo de visión de la cámara PTZ, debido a que el movimiento de la cámara siempre va un paso por detrás, reaccionando al movimiento del objetivo.

Es por eso que se llegó a la conclusión de que era necesario un módulo de control de la cámara móvil que fuese capaz de adelantarse al movimiento que el profesor va a hacer, prediciendo así su futura posición antes de que ocurriese.

Para realizar dicha predicción, de todos los métodos que existen en la literatura, se optó por utilizar uno de los más conocidos que es el filtro de Kalman, resumido brevemente en la sección 2.4, y explicado en detalle en [9]. Se escogió este algoritmo por dos razones fundamentales; la primera de ellas es por los buenos resultados que se obtienen utilizándolo en entornos de seguimiento o predicción de movimiento. La segunda razón es que es un algoritmo que funciona en tiempo real ya que solo utiliza el estado anterior para predecir el actual.

En este trabajo, el filtro de Kalman se utiliza, únicamente, para predecir la posición futura del objetivo, pero no para los demás parámetros de la cámara (velocidad y zoom). Para estos dos se utiliza la misma estrategia que con el sistema de reglas, con la única diferencia de que en este caso se utiliza una posición futura del objetivo, y no la actual (véase la Figura 4.3).

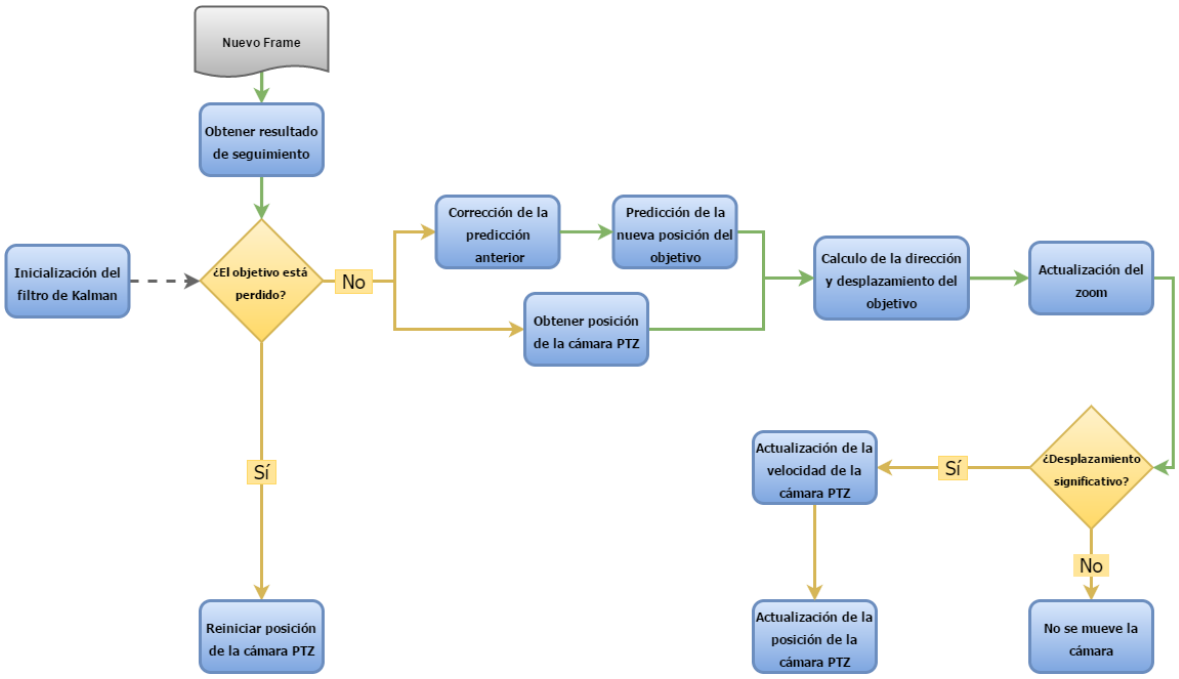


Figura 4.3: Diagrama propuesto para el control de la cámara utilizando el filtro Kalman.

Como se puede ver en la Figura 4.3, lo primero que se hace es inicializar el filtro de Kalman. Esta inicialización se hace una sola vez y al principio del todo, cuando se obtiene el objetivo que se desea seguir (antes de ejecutar el algoritmo de seguimiento), mediante el detector de personas HOG. Aunque es cierto que, realmente, lo que interesa es la posición del objetivo, se utilizan cuatro parámetros que son:

- Posición central del objetivo (coordenada *x* e *y*).
- Velocidad de desplazamiento del objetivo (en el eje '*x*' y en el eje '*y*').

La velocidad de desplazamiento del objetivo es necesaria para poder predecir la futura posición central del objetivo siguiendo el siguiente sistema de ecuaciones:

$$\begin{pmatrix} x(t) \\ y(t) \\ V_x(t) \\ V_y(t) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x(t-1) \\ y(t-1) \\ V_x(t-1) \\ V_y(t-1) \end{pmatrix} + w(t-1)$$

, donde $w(t-1)$ es el modelo de ruido, y además se observa que el estado actual depende, única y exclusivamente, del estado en el instante anterior.

Por lo tanto, una vez inicializado el filtro de Kalman con la posición del profesor y velocidad nula, al igual que con el sistema de reglas, cada vez que llega un *frame* nuevo se obtiene la posición que devuelve el algoritmo de seguimiento, reiniciando la posición de la cámara en caso de haber perdido al objetivo. En caso de no haber perdido al objetivo, se hace una corrección de la predicción anterior sabiendo la posición y velocidad actual, mejorando así futuras predicciones, y seguido de eso se predice la posición del profesor en la siguiente iteración.

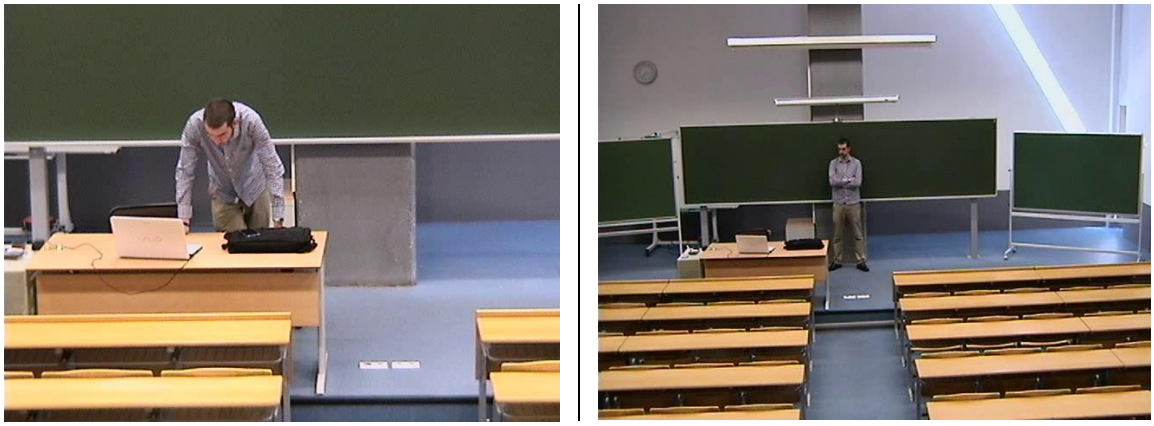
Una vez que se tiene la predicción de la posición en la que estará el profesor, se obtiene también la posición actual de la cámara y se sigue el mismo esquema que cuando se utilizaban reglas para mover la cámara. Primero se calcula el desplazamiento y la dirección en la que se mueve el objetivo. Seguido de eso, se actualiza el zoom y se comprueba si se debe mover la cámara para, finalmente, en caso de tener que mover la cámara, determinar a qué velocidad se debe hacer y, por último, mandar la instrucción a la cámara PTZ.

Utilizando esta aproximación se consigue mejorar el funcionamiento de la producción automática de la cámara PTZ, adelantándose al movimiento del objetivo y contrarrestando el efecto del retardo existente.

4.3 CONTROL DEL ZOOM

En este capítulo se ha hablado de cómo se ha conseguido automatizar el movimiento de la cámara PTZ, controlando la posición a la que apunta y la velocidad a la que se desplaza hasta apuntar a dicha posición, pero también es importante tener un control del zoom, de forma que éste sea suave y fluido y no brusco y molesto para el usuario.

En este trabajo se distinguen tres tipos de perfiles de zoom que se determinan, de manera individual, para cada usuario. Dos de los tres perfiles de zoom son los que se pueden observar en la Figura 4.4. En a) se muestra un zoom cerrado con el objetivo de solo mostrar al profesor, y en b) se tiene un zoom más abierto, obteniendo así una visión más completa de la escena. El tercer perfil de zoom que hay disponible, es un zoom adaptativo, el cual se ajusta al movimiento del profesor en la escena, abriéndose en caso de que éste se mueva, o cerrándose en caso contrario.



a) b)

Figura 4.4: Posibles perfiles de zoom. a) Cerrado. b) Abierto.

Tanto en el diagrama que describía el movimiento de la cámara utilizando reglas (véase Figura 4.2), como en que utiliza el filtro de Kalman (véase Figura 4.3), existe un bloque denominado "*actualización del zoom*", y es que, como se ha dicho anteriormente, para actualizar el zoom, es necesario conocer el movimiento del objetivo que se está siguiendo.

El perfil con zoom adaptativo utiliza tres niveles de zoom (abierto, intermedio y cerrado), empezando con un zoom abierto que muestra gran parte de la escena. Si el objetivo se queda en la misma posición, de forma que no haya que mover la cámara, durante 20 *frames*, el zoom se incrementa, pasando a utilizar un zoom intermedio. Si el objetivo se queda estacionario otros 40 *frames*, el zoom se vuelve a incremente hasta su nivel máximo.

Zoom Actual	Velocidad de la cámara	Zoom final
Abierto	Lenta	Abierto
	Media	
	Rápida	
Intermedio	Lenta	Intermedio
	Media	Abierto
	Rápida	
Cerrado	Lenta	Intermedio
	Media	Abierto
	Rápida	

Tabla 4-1: Zoom de la cámara dependiendo de la velocidad del objetivo.

En el momento que el objetivo se mueve y se determina que hay que mover la cámara, el zoom de la cámara puede tener dos valores zoom dependiendo del actual y de la velocidad a la que se mueve (véase Tabla 4-1). Si la cámara tenía un zoom abierto, no se modifica, manteniendo el zoom. Si, por el contrario, tenía zoom intermedio y el movimiento no es leve, se pasa a un zoom abierto, y si es leve se mantiene el nivel de

zoom. Finalmente, si tenía un zoom cerrado y hay movimiento leve, se pasa a un zoom intermedio, y sino a zoom abierto.

I)

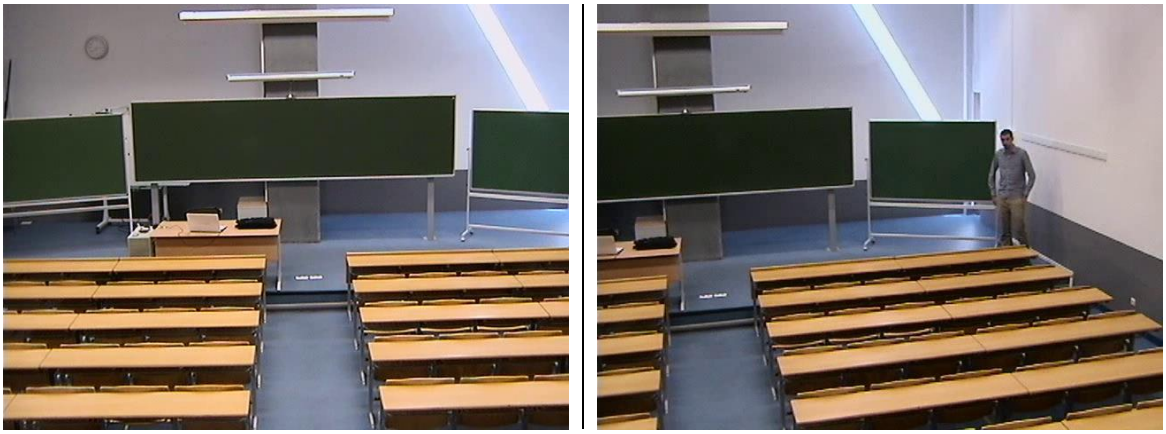
II)

III)

IV)

V)

VI)



VII) VIII)

Figura 4.5: Secuencia de la cámara PTZ con un perfil de zoom adaptativo.

El la Figura 4.5 Se observa la secuencia del funcionamiento del algoritmo completo, después de haberlo inicializado, utilizando un perfil de zoom adaptativo. En la imagen I) se observa que empieza con un zoom abierto, pasa a un zoom intermedio en II) y, luego a un zoom cerrado en III) ya que el sujeto se queda estático, es decir, se puede mover, pero no lo suficiente como para mover la cámara. A continuación, en la imagen IV), el sujeto se desplaza ligeramente hacia un lateral y, por lo tanto, como se mostraba en la Tabla 4-1, se pasa a un zoom intermedio. En la imagen V), el sujeto se sigue moviendo, pero esta vez a más velocidad, de forma que se pasa a un zoom abierto, siguiendo con lo descrito en la tabla anterior. En la imagen VI) el sujeto pasa por detrás de la pizarra y, por lo tanto, el algoritmo de seguimiento determina que ha perdido al objetivo de manera que la cámara se reinicia al punto central de la escena con un zoom abierto, como se puede ver en VII). Finalmente, el algoritmo vuelve a encontrar al sujeto en el lado derecho de la escena y la cámara PTZ vuelve a enfocarlo.

Capítulo 5. EVALUACIÓN Y PRUEBAS

Una vez terminado el desarrollo del algoritmo se ha procedido a hacer pruebas sobre el mismo midiendo el tiempo de ejecución y los resultados que se obtienen utilizando el detector de personas en la inicialización. También se ha hecho una evaluación del módulo que recupera al objetivo, haciendo una comparación con el módulo anterior con el fin de comprobar si la integración del detector de personas mejora los resultados que se obtenían. Finalmente, se ha evaluado el control de la cámara comparándolo con los resultados que se obtenían en [1].

Para la obtención de las medidas cuantitativas de tiempo de procesado, dato de especial relevancia en un sistema de tiempo real, se ha utilizado un equipo con un procesador Intel® Core™ i7-3632QM con una CPU a 2.2 GHz y 6GB de RAM. El sistema operativo es Windows 10 de 64 bits.

El algoritmo original fue implementado en C++ con OpenCV versión 2.4.1, y las modificaciones que se han hecho en este trabajo han sido implementadas en C++ con OpenCV versión 2.4.11.

5.1 DESCRIPCIÓN DEL *DATASET*

Antes de explicar las diferentes pruebas que se han realizado y los resultados que se han obtenido, es necesario detallar el *dataset* o conjunto de secuencias sobre las que se han realizado dichas pruebas.

El primer *dataset* está compuesto por un total hay 15 secuencias de vídeo, captadas por la cámara fija, de distintas duraciones, con diferentes situaciones que pueden complicar el seguimiento del objetivo. Dichos problemas incluyen, pero no se limitan, a: el objetivo se da la vuelta, oclusión parcial del objetivo, el objetivo abandona la escena y vuelve a entrar, etc. Para más información acerca de este *dataset*, en el anexo A se detalla la duración de cada secuencia, el número de *frames* que tiene y los distintos problemas que se incluyen.

Para evaluar el número de detecciones, por *frame*, que se obtienen utilizando los distintos detectores, se ha creado otro *dataset* de detección, compuesto a partir de *frames* extraídos de las secuencias que componen el *dataset* anterior. Las imágenes que componen este nuevo *dataset* se muestran en el anexo B.

Es importante mencionar que siempre se consigue un 100% de acierto en la inicialización del objetivo, dado que las imágenes que componen el *dataset* de detección son muy sencillas ya que se parte de la asunción que siempre hay una persona en la escena y sólo hay una persona.

5.2 EVALUACIÓN DE LA INICIALIZACIÓN

Como se ha comentado anteriormente, en este trabajo se buscaba inicializar el algoritmo de seguimiento de forma automática, sin necesidad de tener a un operario que indicase la posición inicial del profesor. Además, este proceso no tenía la restricción de tener que funcionar en tiempo real, ya que hasta que no se indica dónde está el profesor, no se ejecuta el módulo de seguimiento.

Como se ha explicado en la sección 3.1, esta inicialización consta de tres detectores, uno de persona, otro de cara frontal y otro de cara de perfil, con el objetivo de obtener una sola *bounding box* que contenga al profesor.

Tras realizar pruebas en múltiples secuencias, se ha medido el número medio de *frames* necesarios para obtener la posición del profesor y el número medio de detecciones que se obtienen por cada *frame* (véase Tabla 5-1). Se puede observar que, como se explicó en la sección 3.1, la utilización del detector de personas devuelve, de media, más de una detección por cada *frame*. Con la introducción de los detectores de cara, el número de detecciones por *frame* disminuye notablemente, pero se garantiza que el objeto detectado será el profesor.

En la Tabla 5-1 también se muestra el número medio de *frames* que se analizan hasta obtener al menos una detección. Visto lo descrito en el párrafo anterior, no es sorprendente que, al utilizar el detector de personas, solo se necesite un *frame*, de media, para obtener al menos una detección, pero como también se describió en el párrafo anterior, de media se obtiene más de una detección, por lo tanto, es necesario distinguir entre las detecciones para saber cuál es la que realmente contiene una persona.

	Detector de personas	*Detector de personas y de cara frontal*	*Detector de personas, de cara frontal y de cara de perfil*
Número medio de detecciones por frame	1.19	0.73	0.77
Número medio de frames para detectar	1	1.86	1.85

Tabla 5-1: Evaluación del número medio de frames necesarios para detectar al profesor.

En cambio, incorporando al detector de personas los detectores de caras, frontal y de perfil, o solamente el detector de caras frontales, se obtienen resultados muy similares, necesitando de media casi dos *frames* para obtener al menos una detección. La gran diferencia que existe entre utilizar múltiples detectores es que se obtiene una

detección correcta del profesor, distinguiendo así entre todas las posibles detecciones que devolvía utilizar únicamente el detector de personas.

Además de evaluar los resultados obtenidos con la inicialización automática, y a pesar de que no es necesario que funcione en tiempo real la inicialización, se ha decidido medir el tiempo que tarda en ejecutarse para comprobar que el tiempo no es excesivo (véase Tabla 5-2).

	Detector de personas	*Detector de personas y de cara frontal*	*Detector de personas, de cara frontal y de cara de perfil*
Tiempo medio de ejecución por frame (segundos)	0.238	0.243	0.248

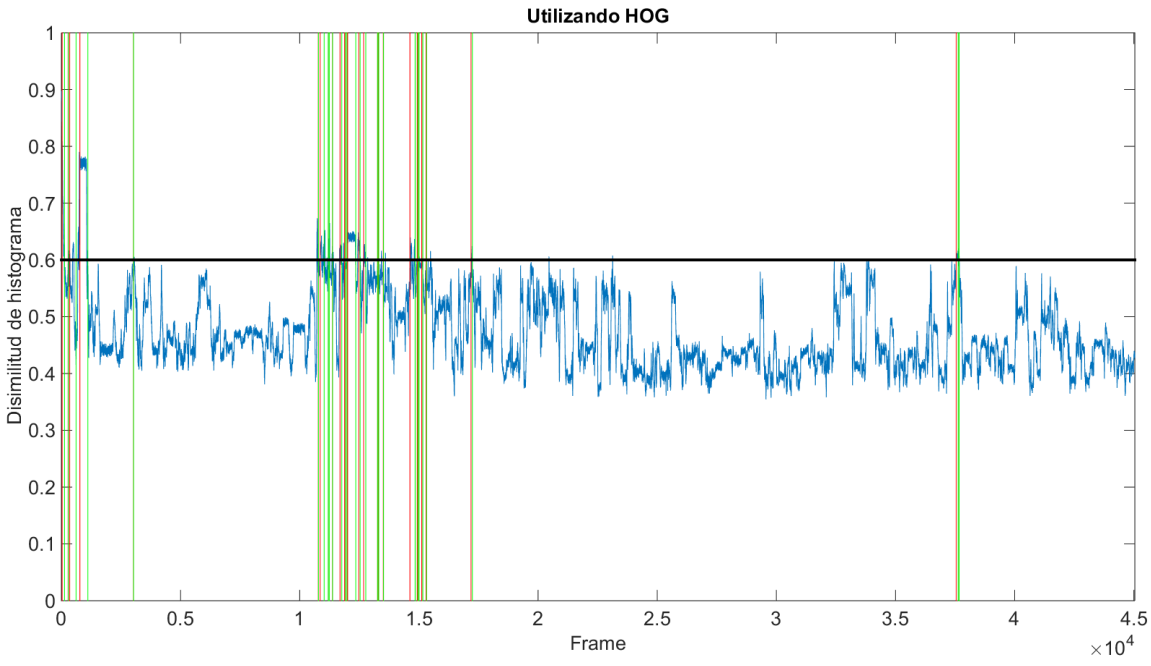
Tabla 5-2: Evaluación del tiempo de ejecución.

Como se puede ver en la Tabla 5-2, el tiempo que se tarda en analizar un *frame* es despreciable para una aplicación que no exige tiempo real, incluso cuando se incluyen los dos detectores de caras, de forma que, desde el momento en que se pone en marcha, hasta que el algoritmo empieza a seguir al profesor sería casi instantáneo. Además, este método es incluso más veloz que el método manual, dependiendo de la exactitud que quisiese el usuario.

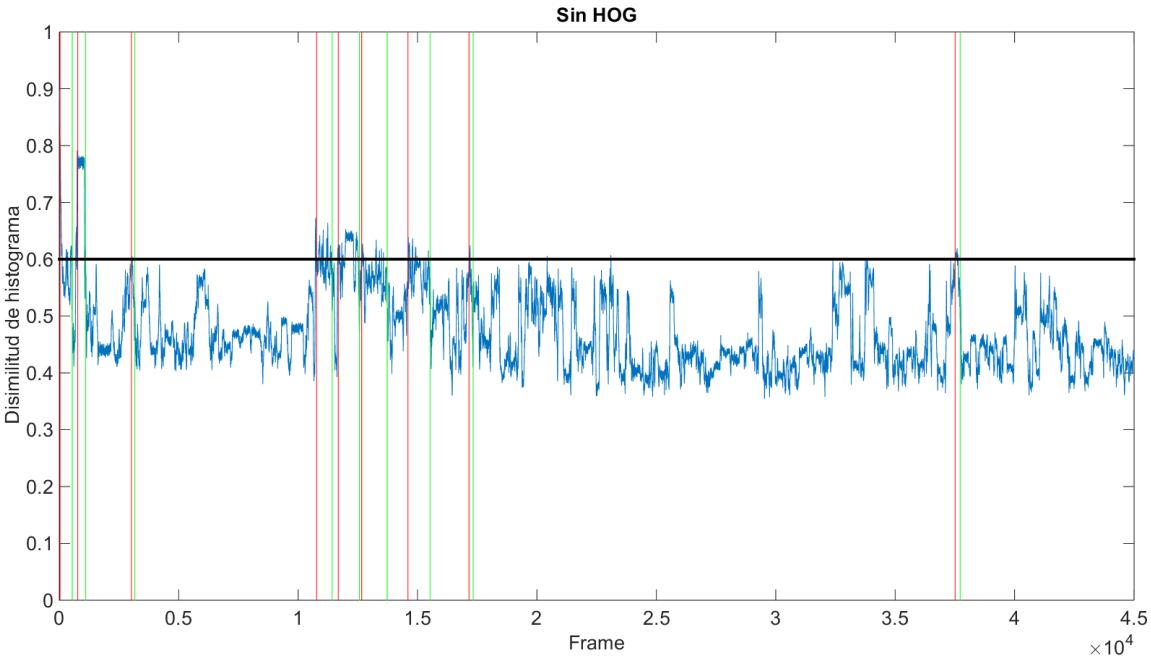
5.3 EVALUACIÓN DE LA RECUPERACIÓN DEL OBJETIVO

Finalizada la evaluación de la inicialización del objetivo, se procede a evaluar la recuperación del mismo cuando el algoritmo de seguimiento determina que lo ha perdido. Para ello se va a comparar el módulo encargado de la recuperación del objetivo utilizando y sin utilizar el detector de personas HOG en una secuencia real de larga duración (30 minutos).

En la Figura 5.1, se muestra la progresión que tiene el algoritmo de seguimiento a lo largo de una secuencia. En azul se observa la disimilitud de entre el histograma y el modelo por cada *frame*, mientras que la línea negra horizontal es el umbral que ayuda a determinar si la *bounding box* actual contiene, o no, al profesor.



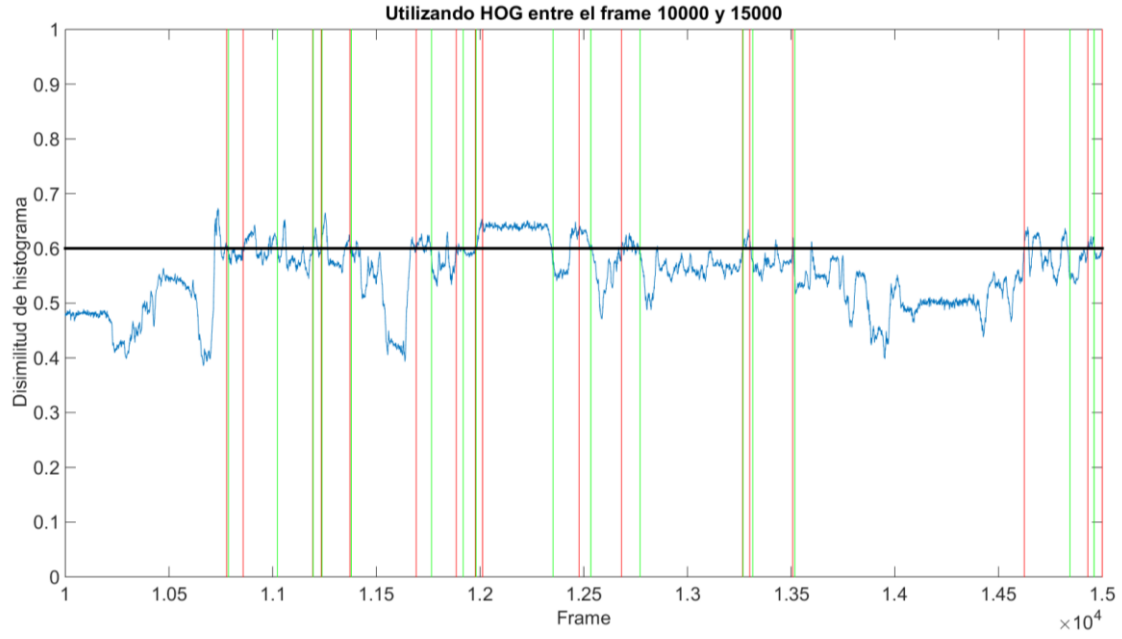
a)



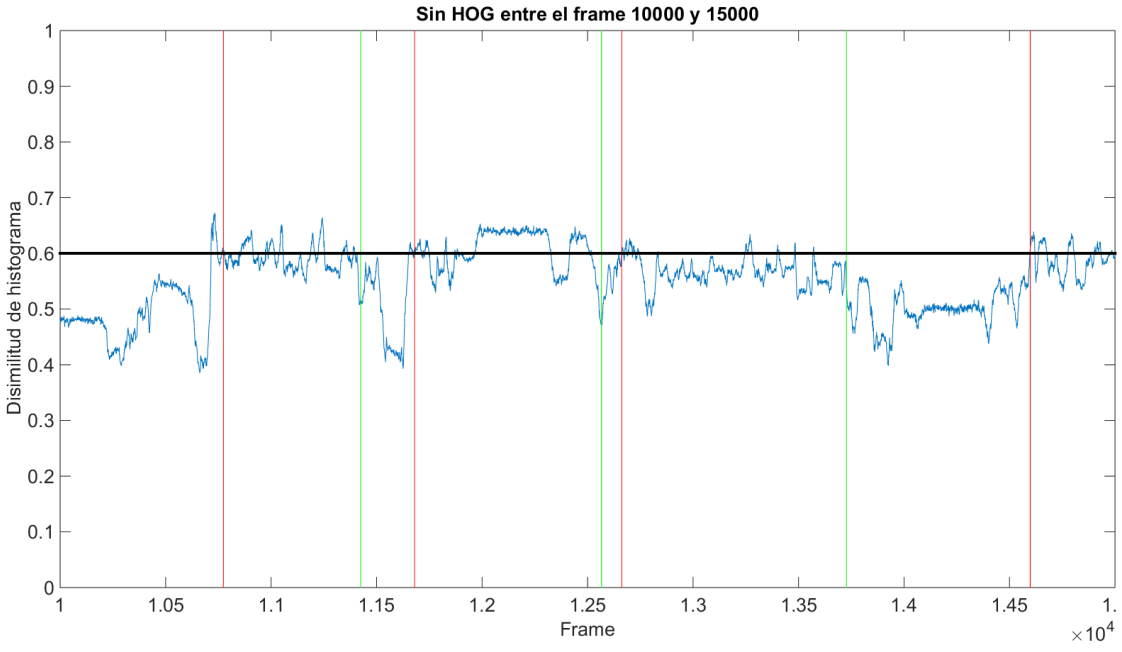
b)

Figura 5.1: Comparación de la recuperación del objetivo: a) utilizando el detector de personas HOG, b)Sin utilizar el detector de personas HOG.

En la misma figura, las líneas rojas verticales indican cuando el algoritmo considera que ha perdido al profesor, mientras las verdes indican que se ha recuperado al objetivo, de forma que se obtiene una recuperación por cada pérdida del objetivo (ver detalle en la Figura 5.2).



a)

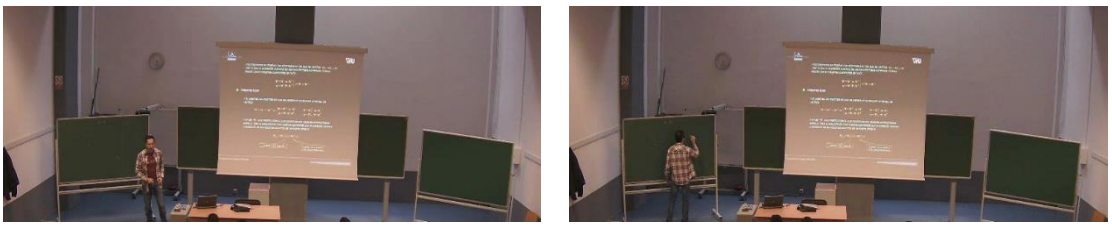


b)

Figura 5.2: Zoom de las imágenes de la Figura 5.1 entre los frames 10000 y 15000.

De las figuras anteriores se puede ver claramente que la utilización del detector de personas HOG da lugar a muchas más recuperaciones, aunque también es cierto que se obtienen muchas más pérdidas, sobre todo entre el *frame* 10000 y 15000 (véase Figura 5.2). En ese tramo de la secuencia, el profesor se gira repetidamente para escribir en la pizarra (véase Figura 5.3). La razón por la cual el algoritmo de seguimiento le pierde, es debido a que el modelo se generó con el profesor mirando a la cámara y por lo tanto, el histograma es ligeramente diferente al que se obtiene con él de espaldas. A pesar de ello, el modelo se va actualizando y como se puede ver en la Figura 5.1, la

disimilitud va decayendo hasta estar por debajo del umbral entre los *frames* 15000 y 25000



a) b)

Figura 5.3:Distintas posturas del profesor.

Como se explicó en la sección 3.4, el sistema existente de recuperación del objetivo requería de que éste se moviese, obteniendo puntos en aquellas zonas donde hay movimiento, pero en este vídeo, entre los *frames* 10000 y 15000, cuando el profesor escribe en la pizarra, no se genera suficiente movimiento y por lo tanto no es capaz de recuperar al objetivo.

En cambio, la utilización del detector de personas HOG no requiere de este movimiento. A estas detecciones se les calcula el histograma y se compara con el modelo, de forma que, si la disimilitud está por debajo del umbral, se concluye que es el objetivo que se había perdido.

	Utilizando el detector de personas HOG	*Sin utilizar el detector de personas HOG*
Número de veces que el objetivo se pierde y se recupera	27	9
Número de frames que el objetivo está perdido	1992	4901
Porcentaje de frames que el objetivo está perdido	4,42%	10,89%

Tabla 5-3: Comparación entre utilizar o no el detector de personas HOG en la recuperación del objetivo.

Finalmente, se decidió comparar cuantitativamente la diferencia que existía entre utilizar el detector de personas HOG, o no usarlo, para recuperar al objetivo. Observando la Tabla 5-3, se confirma lo que se apreciaba en la Figura 5.1, al utilizar el detector de personas HOG, el objetivo se recupera más veces. De forma más importante, esto produce que, como se puede ver en la misma tabla, el número de *frames* totales de la secuencia en los que el objetivo está perdido se reduce por más de la mitad, confirmando que la utilización del detector de personas mejora notablemente la recuperación del objetivo.

5.4 EVALUACIÓN DEL MOVIMIENTO DE LA CÁMARA

Una vez evaluada la inicialización del objetivo y el algoritmo de seguimiento, se procede a evaluar el módulo de movimiento de la cámara PTZ y las dos soluciones que han sido propuestas en este trabajo para hacer dicha función. Este módulo es importante ya que el resultado que se obtiene con él es la imagen final que verá el usuario

Esta evaluación no se puede realizar de forma objetiva ya que no se dispone de un *ground truth* con el que comparar los resultados que se obtienen. Lo que sí se puede hacer es una evaluación subjetiva, comparando entre las dos soluciones propuestas los resultados que se obtienen.

Utilizando el sistema basado en reglas que se explicaba en la sección 4.1, la producción final tiene una calidad muy buena en general. El movimiento de la cámara en vertical y horizontal es suave y fluido y se ajusta a la velocidad a la que se desplaza el profesor. Está solución tiene el problema de que siempre reacciona a lo que hace el objetivo, por lo que es posible que, si se desplaza a una velocidad alta, el sujeto pueda llegar a salir de la imagen que capta la cámara PTZ.

Al igual que la propuesta anterior, con la solución que hace uso del filtro de Kalman también se obtiene una calidad de producción muy buena. Al igual que con la otra solución, el movimiento en vertical y horizontal es suave y fluido. Además, esta solución consigue adelantarse al movimiento del objetivo, consiguiendo así un mejor resultado cuando se mueve a alta velocidad. El único problema ocurre cuando el zoom está muy cerrado y el objetivo se mueve a una velocidad alta, causando que sea posible que desaparezca de la imagen.

Ambas soluciones también comparten un contratiempo que ocurre cuando se utiliza el zoom. La cámara PTZ utilizada en este trabajo ofrece ocho velocidades de zoom, pero incluso utilizando la velocidad más lenta, en el instante en el que se cambia de zoom, ocurre un primer movimiento brusco.

También se tienen problemas en ambas soluciones si el algoritmo tarda mucho en detectar que ha perdido al sujeto. Normalmente, cuando el algoritmo pierde al objetivo, su posición en la escena se mantiene estática, por lo tanto, si tarda demasiado en detectar que ha perdido al sujeto, la cámara hará zoom en esa posición.

En general, el movimiento de la cámara resultante tras la implementación de ambas soluciones propuestas mejora notablemente los resultados que se obtenían con el algoritmo existente. Entre ellas, el funcionamiento de ambas es muy bueno, consiguiendo un movimiento suave y fluido de la cámara PTZ. Aun así, el módulo que hace uso del filtro de Kalman funciona ligeramente mejor, adelantándose al movimiento del sujeto que se desea seguir.

Capítulo 6. CONCLUSIONES Y TRABAJO FUTURO

Tras detallar todo lo que se ha implementado en este trabajo y realizar la evaluación de dicha implementación, en este capítulo se procede a exponer las conclusiones y proponer futuras vías de trabajo.

6.1 CONCLUSIONES

En este trabajo se ha buscado incrementar un paso más la automatización del seguimiento del profesor en un aula para la emisión de clases, mejorando el trabajo de [1], mediante el uso del detector de personas HOG, y creando un nuevo módulo para el manejo de la cámara PTZ. Tras todo el desarrollo y la evaluación del algoritmo, se puede llegar a varias conclusiones.

La solución propuesta es válida: en este trabajo se ha conseguido cumplir los objetivos principales que se marcaban en el capítulo 1 de esta memoria.

Se ha conseguido automatizar la inicialización del objetivo haciendo uso de varios detectores para obtener su posición, sustituyendo así al método original que requería de un técnico, o del propio profesor, para marcar la posición del objetivo a seguir.

Se ha mejorado el esquema de reglas que controlaban el movimiento de la cámara PTZ, aportando dos posibles soluciones, una basada en reglas y otra que además hace uso del filtro de Kalman. Con ambas soluciones se obtienen resultados muy buenos, pero con la implementación que usa el filtro de Kalman se tiene una ligera ventaja ya que se anticipa al movimiento del profesor.

Además, se ha conseguido mejorar el módulo de recuperación del objetivo haciendo uso del mismo detector de personas que se utiliza en la inicialización, obteniendo así mejores resultados.

6.2 TRABAJO FUTURO

Debido a la duración limitada de este trabajo, hay aspectos que no se han podido llevar a cabo y que se pueden hacer en un futuro para mejorar el funcionamiento general del algoritmo.

En primer lugar, la inicialización del profesor se podría mejorar haciendo uso de algún método que solo seleccionase a él, sin nada de fondo, de forma que el modelo generado se ajustase lo más posible al profesor consiguiendo que el histograma describiese, exclusivamente, al profesor.

Sería interesante, también, mejorar la parte del algoritmo de seguimiento que determina si se ha perdido o no al sujeto que inicialmente se estaba siguiendo. Esto se haría para buscar reducir el tiempo que la cámara móvil está enfocando a una parte de la escena que no incluye al profesor.

Una última vía de trabajo futuro es comprobar que, en otras aulas, con otras condiciones de iluminación y con una escena distinta, el algoritmo se sigue comportando de igual forma que en el aula 6 de la Escuela Politécnica Superior.

REFERENCIAS

[1] Gónzalez Huete, A., & Bescós Cano, J. (2013). *Seguimiento y producción automática mediante cámaras PTZ en entornos red.*

[2] Lucas, B. D., & Kanade, T. (1981, August). An iterative image registration technique with an application to stereo vision. In *IJCAI* (Vol. 81, pp. 674-679).

[3] Shi, J., & Tomasi, C. (1994, June). Good features to track. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94., 1994 IEEE Computer Society Conference on* (pp. 593-600). IEEE.

[4] Isard, M., & Blake, A. (1998). Condensation—conditional density propagation for visual tracking. *International journal of computer vision*, *29*(1), 5-28.

[5] Arulampalam, M. S., Maskell, S., Gordon, N., & Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *Signal Processing, IEEE Transactions on*, *50*(2), 174-188.

[6] Comaniciu, D., Ramesh, V., & Meer, P. (2000). Real-time tracking of non-rigid objects using mean shift. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on* (Vol. 2, pp. 142-149). IEEE.

[7] Ukrainitz, Y., & Sarel, B. (2004). Mean shift: Theory and applications.*Weizmann Institute of Science, http://www. wisdom. weizmann. ac. il/~ vision/courses/2004_2/files/mean_shift/mean_shift. ppt*.

[8] Dalal, N., & Triggs, B. (2005, June). Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* (Vol. 1, pp. 886-893). IEEE.

[9] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, *82*(1), 35-45.

[10] Singer, R. A. (1970). Estimating optimal tracking filter performance for manned maneuvering targets. *Aerospace and Electronic Systems, IEEE Transactions on*, (4), 473-483.

[11] Rubio Redondo, Á. J. (2015). Herramientas de apoyo a la emisión de clases presenciales.

[12] Ning, J., Zhang, L., Zhang, D., & Wu, C. (2012). Robust mean-shift tracking with corrected background-weighted histogram. *Computer Vision, IET*, *6*(1), 62-69.

[13] Sanjuán García, J. (2014). Seguimiento de objetos en tiempo real.

ANEXOS

A. DESCRIPCIÓN DETALLADA DEL *DATASET DE SECUENCIAS DE VÍDEO*

	Duración	*Numero de frames*	*Tipo de problema*
Secuencia 1	00:00:20	163	• El objetivo se da la vuelta. • El objetivo se pone de lado. • Movimiento lento del objetivo.
Secuencia 2	00:00:19	156	• Oclusión parcial. • El objetivo se da la vuelta. • Movimiento lento del objetivo.
Secuencia 3	00:05:02	2416	• Oclusión de las piernas. • El objetivo se pone de lado. • El objetivo se da la vuelta. • Movimiento lento y rápido del objetivo.
Secuencia 4	00:02:02	3069	• El objetivo se pone de lado. • El objetivo se da la vuelta. • Movimiento lento y rápido del objetivo. • Mesas y asientos de los estudiantes caben en el campo de visión de la cámara.
Secuencia 5	00:01:59	2992	• El objetivo se pone de lado. • El objetivo se da la vuelta. • Mesas y asientos de los estudiantes caben en el campo de visión de la cámara. • Movimiento lento del objetivo. • El objetivo se quita el jersey.
Secuencia 6	00:02:07	3188	• El objetivo se pone de lado. • El objetivo se da la vuelta. • Mesas y asientos de los estudiantes caben en el campo de visión de la cámara. • Movimiento lento y rápido del objetivo. • El objetivo abandona la escena por el lado derecho y reaparece por el izquierdo.

Secuencia 7	00:01:55	2898	• El objetivo se pone de lado. • El objetivo se da la vuelta. • Mesas y asientos de los estudiantes caben en el campo de visión de la cámara. • Movimiento lento y rápido del objetivo. • El objetivo abandona la escena por el lado derecho y reaparece por el mismo lado.
Secuencia 8	00:03:10	4757	• El objetivo se pone de lado. • El objetivo se da la vuelta. • Mesas y asientos de los estudiantes caben en el campo de visión de la cámara. • Movimiento lento y rápido del objetivo. • El objetivo abandona la escena por el lado derecho y reaparece por el centro con un cambio de vestimenta.
Secuencia 9	00:30:00	45000	• El objetivo se pone de lado. • El objetivo se da la vuelta. • Movimiento lento y rápido del objetivo. • La luces del aula están parcialmente apagadas. • El objetivo abandona la escena por el lado izquierdo y reaparece por el mismo lado. • El objetivo es iluminado por el proyector.
Secuencia 10	00:00:12	217	• El objetivo se pone de lado. • Movimiento lento y rápido del objetivo.
Secuencia 11	00:00:14	292	• El objetivo se pone de lado. • El objetivo se queda estático. • Movimiento lento del objetivo. • El objetivo abandona la escena por el lado izquierdo y reaparece por el mismo lado.
Secuencia 12	00:00:06	175	• Movimiento lento y rápido del objetivo. • El objetivo se queda estático.

Secuencia 13	00:00:24	527	• El objetivo se pone de lado. • El objetivo se da la vuelta. • Movimiento lento y rápido del objetivo. • El objetivo se queda estático. • El objetivo abandona la escena por el lado izquierdo y reaparece por el centro.
Secuencia 14	00:00:57	1442	• El objetivo se pone de lado. • El objetivo se da la vuelta. • Movimiento lento y rápido del objetivo. • El objetivo se queda estático. • El objetivo pasa por detrás de las pizarras y reaparece en el otro lado de la escena.
Secuencia 15	00:00:25	479	• El objetivo se pone de lado. • Movimiento lento y rápido del objetivo. • El objetivo se queda estático. • El objetivo pasa por detrás de las pizarras y reaparece en el otro lado de la escena.

B. *DATASET* DE DETECCIÓN